



Uso de Funciones de Una Sola Fila para Personalizar la Salida

ORACLE®

Copyright © 2004, Oracle. Todos los derechos reservados.

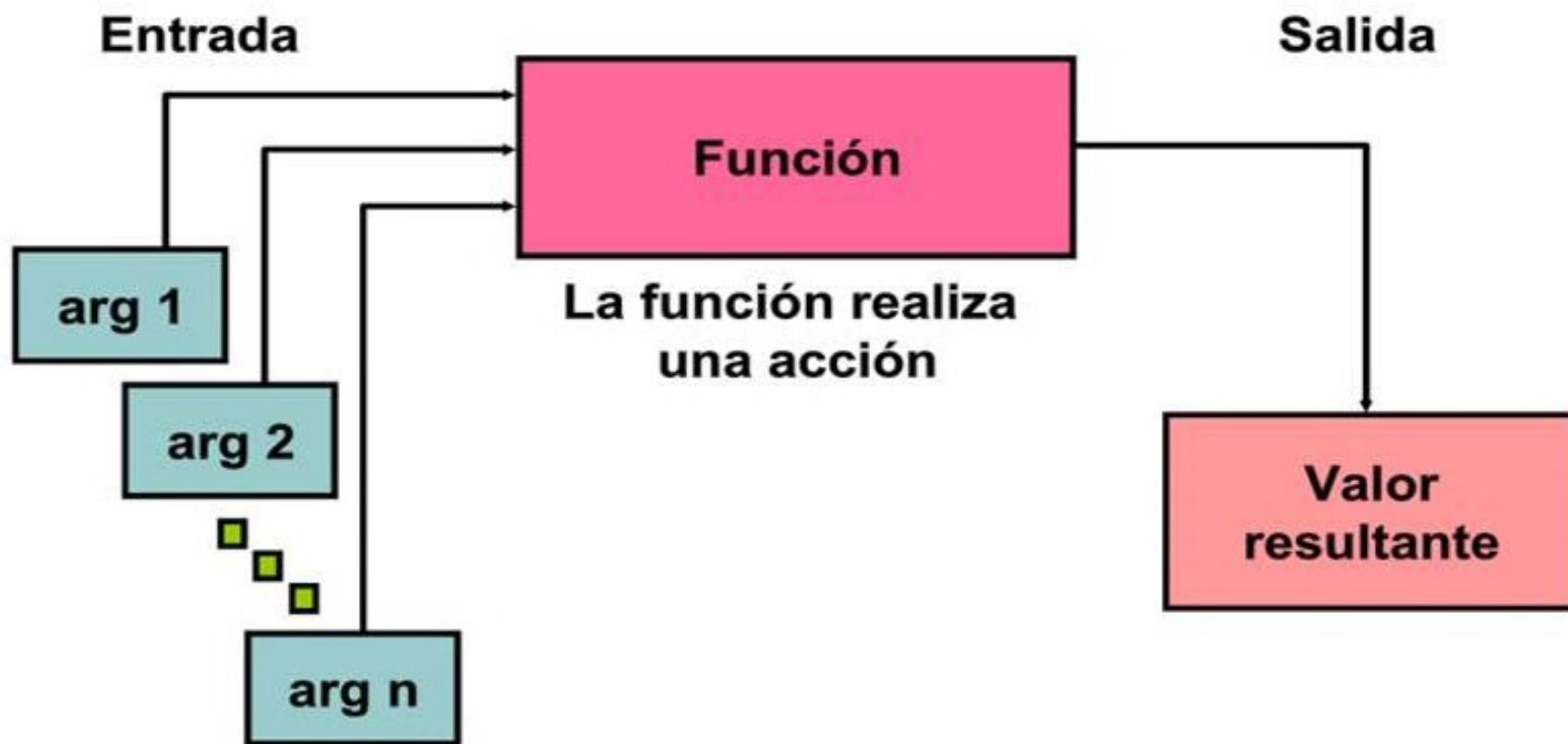
Objetivos

Al finalizar esta lección, debería estar capacitado para:

- **Describir varios tipos de funciones que hay disponibles en SQL**
- **Utilizar funciones de carácter, numéricas y de fecha en sentencias SELECT**
- **Describir el uso de las funciones de conversión**

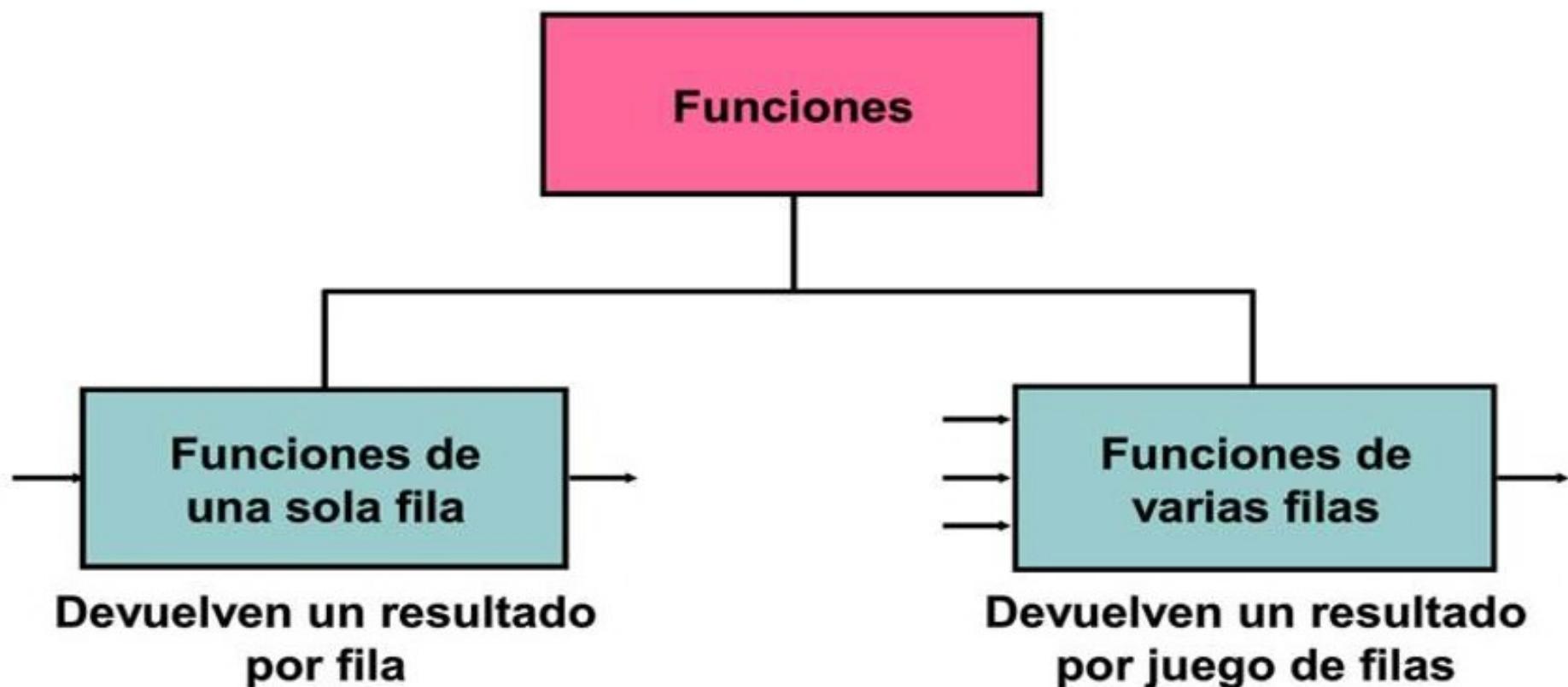


Funciones SQL



ORACLE

Dos Tipos de Funciones SQL



ORACLE

Funciones de una Sola Fila

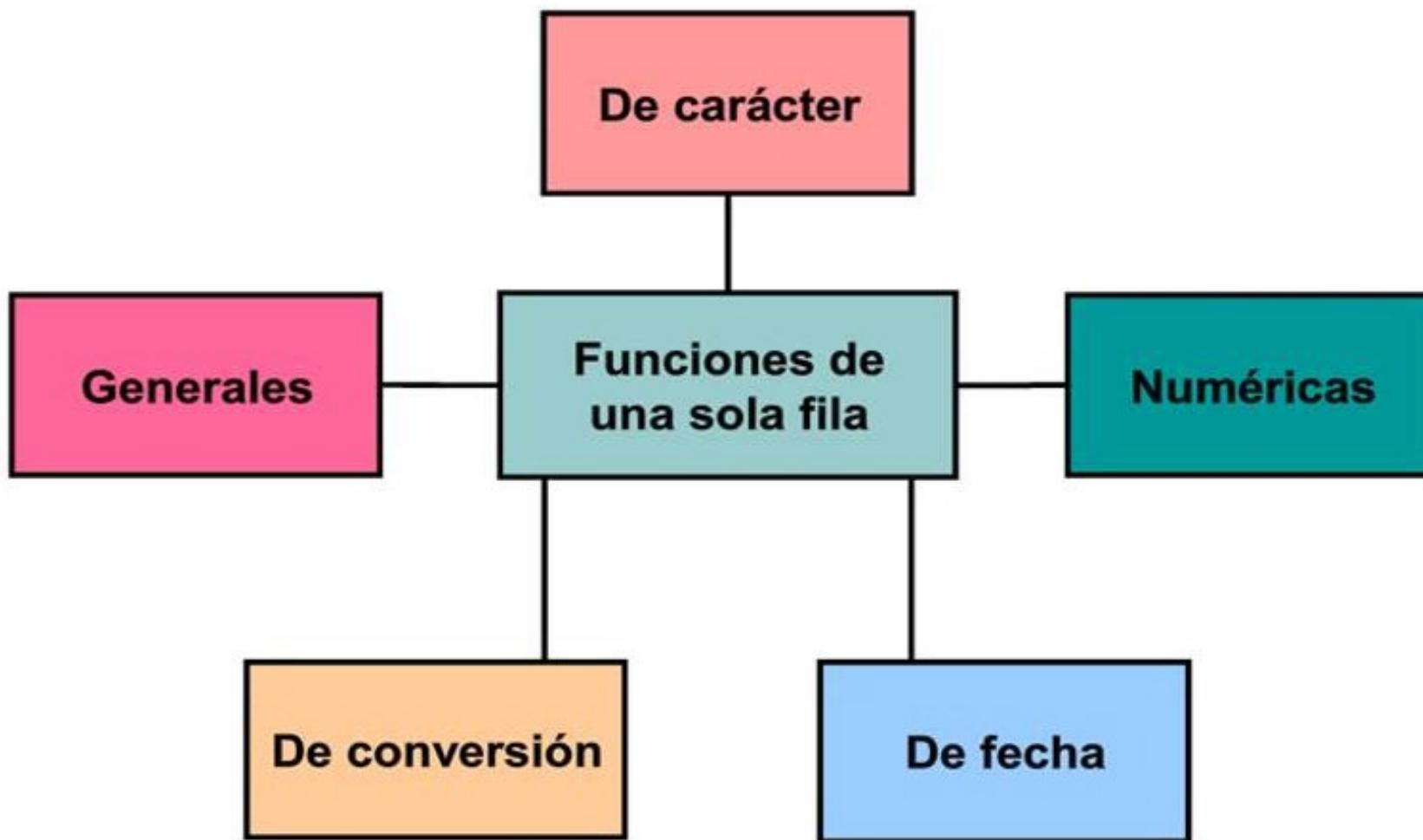
Las funciones de una sola fila:

- Manipulan elementos de datos
- Aceptan argumentos y devuelven un valor
- Actúan en cada fila que se devuelve
- Devuelven un resultado por fila
- Pueden modificar el tipo de datos
- Se pueden anidar
- Aceptan argumentos que pueden ser una columna o una expresión

```
function_name [ (arg1, arg2, . . .) ]
```

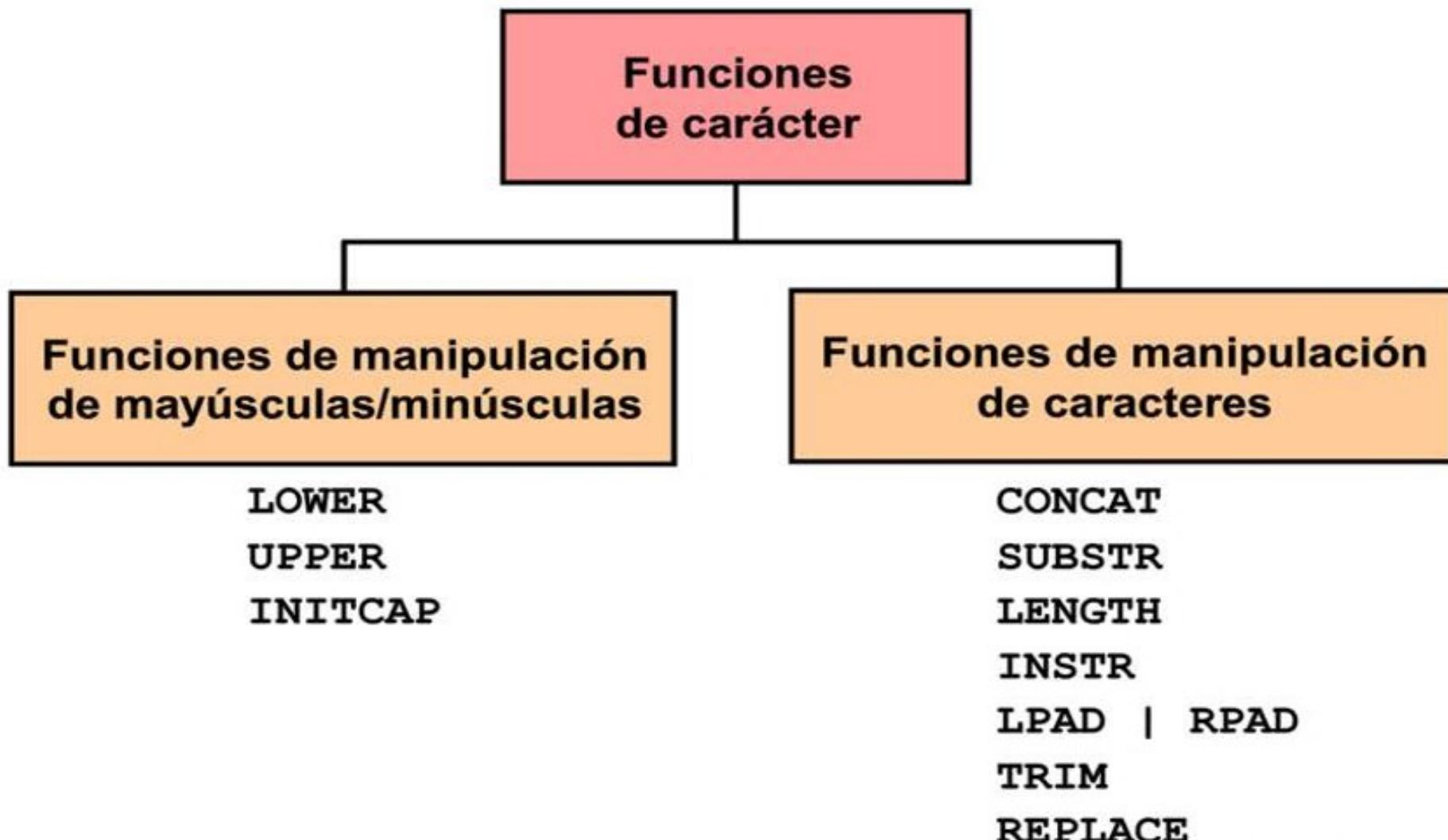
ORACLE

Funciones de una Sola Fila



ORACLE

Funciones de Carácter



ORACLE

Funciones de Manipulación de Mayúsculas/Minúsculas

Estas funciones convierten en mayúsculas y minúsculas las cadenas de caracteres:

Función	Resultado
LOWER ('SQL Course')	sql course
UPPER ('SQL Course')	SQL COURSE
INITCAP ('SQL Course')	Sql Course



Uso de Funciones de Manipulación de Mayúsculas/Minúsculas

Muestre el número de empleado, nombre y número de departamento del empleado Higgins:

```
SELECT employee_id, last_name, department_id  
FROM   employees  
WHERE  last_name = 'higgins';  
no rows selected
```

```
SELECT employee_id, last_name, department_id  
FROM   employees  
WHERE  LOWER(last_name) = 'higgins';
```

EMPLOYEE_ID	LAST_NAME	DEPARTMENT_ID
205	Higgins	110



Funciones de Manipulación de Caracteres

Estas funciones manipulan cadenas de caracteres:

Función	Resultado
CONCAT ('Hello', 'World')	HelloWorld
SUBSTR ('HelloWorld', 1, 5)	Hello
LENGTH ('HelloWorld')	10
INSTR ('HelloWorld', 'W')	6
LPAD (salary, 10, '*')	*****24000
RPAD (salary, 10, '*')	24000*****
REPLACE ('JACK and JUE', 'J', 'BL')	BLACK and BLUE
TRIM ('H' FROM 'HelloWorld')	elloWorld

Uso de Funciones de Manipulación de Caracteres

```
SELECT employee_id, CONCAT(first_name, last_name) NAME,  
      job_id, LENGTH(last_name),  
      INSTR(last_name, 'a') "Contains 'a'?"  
FROM employees  
WHERE SUBSTR(job_id, 4) = 'REP';
```

EMPLOYEE_ID	NAME	JOB_ID	LENGTH(LAST_NAME)	Contains 'a'?
174	EllenAbel	SA_REP	4	0
176	JonathonTaylor	SA_REP	6	2
178	KimberelyGrant	SA_REP	5	3
202	PatFay	MK_REP	3	2

1

2

3

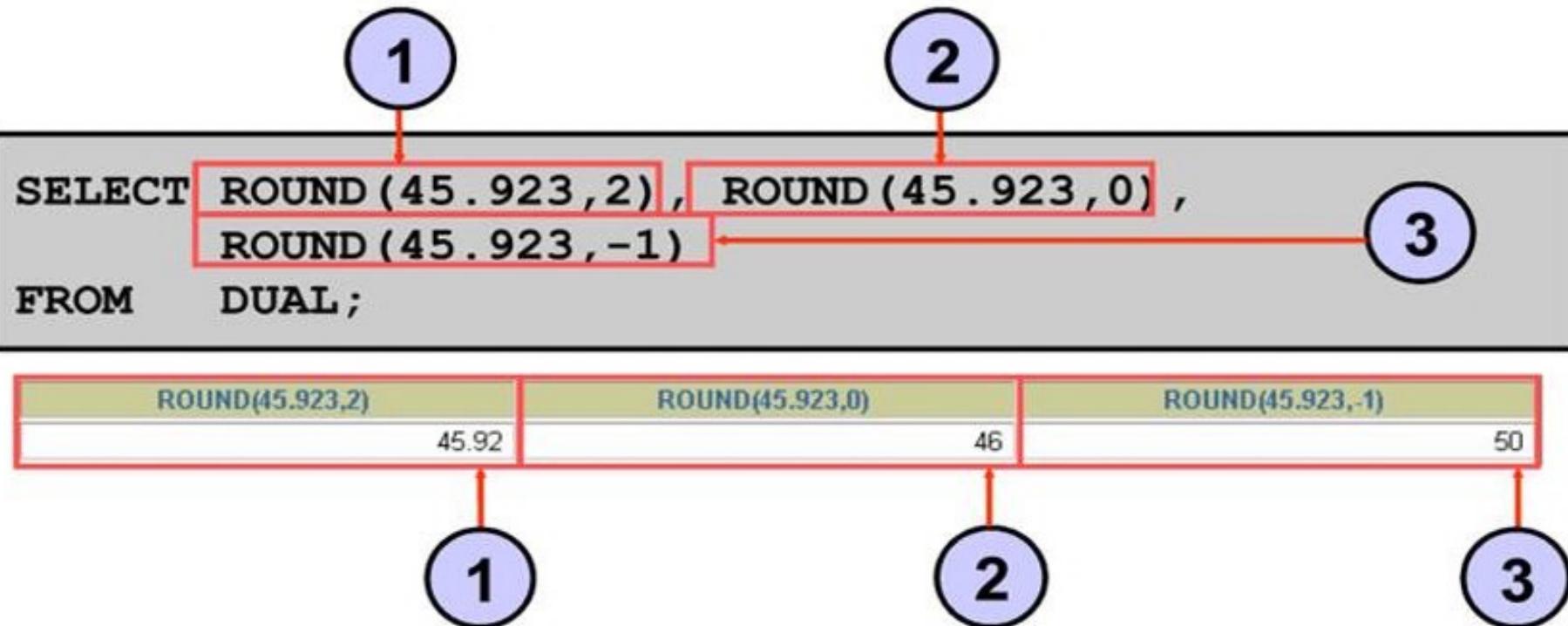
ORACLE

Funciones Numéricas

- **ROUND:** Redondea el valor al decimal especificado
- **TRUNC:** Trunca el valor al decimal especificado
- **MOD:** Devuelve el resto de la división

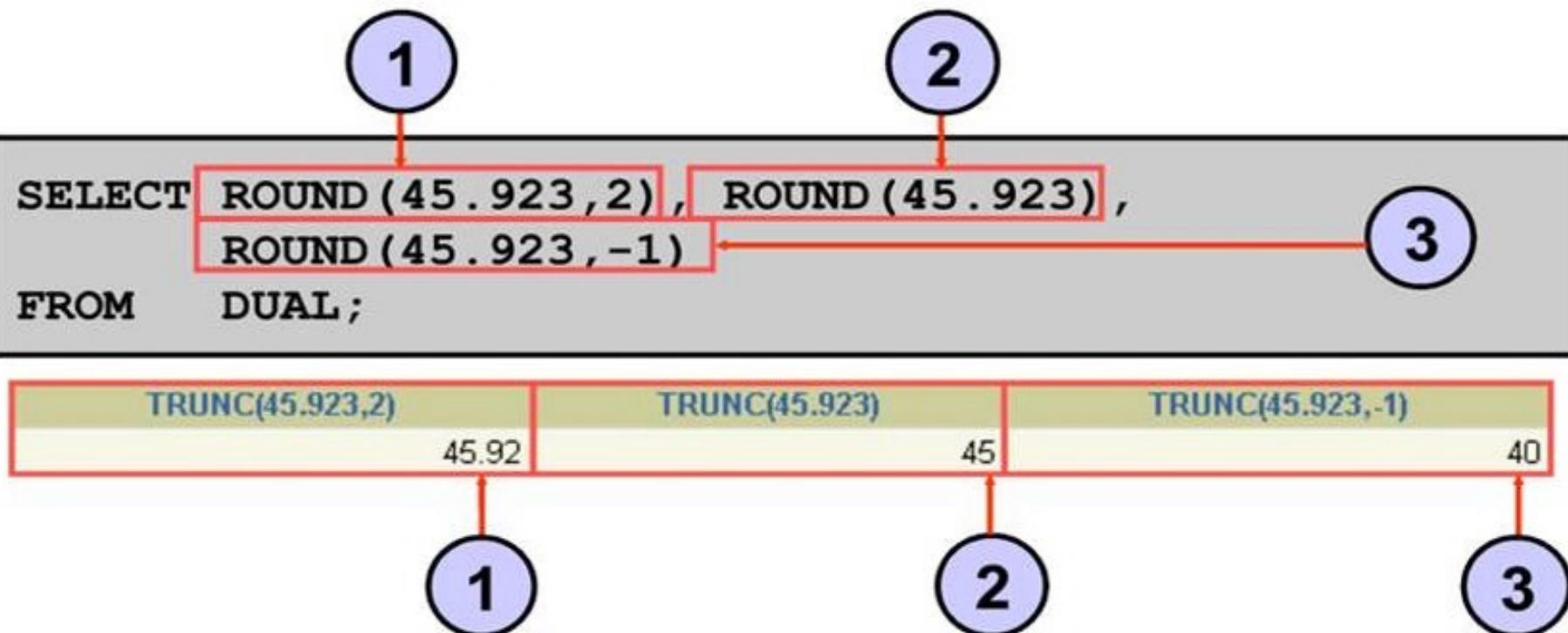
Función	Resultado
ROUND (45.926, 2)	45.93
TRUNC (45.926, 2)	45.92
MOD (1600, 300)	100

Uso de la Función ROUND



DUAL es una tabla ficticia que puede utilizar para ver los resultados de funciones y cálculos.

Uso de la Función TRUNC



ORACLE

Uso de la Función MOD

Para todos los empleados con cargo de representante de ventas, calcule el resto del salario una vez dividido por 5.000.

```
SELECT last_name, salary, MOD(salary, 5000)  
FROM employees  
WHERE job_id = 'SA REP';
```

LAST_NAME	SALARY	MOD(SALARY,5000)
Abel	11000	1000
Taylor	8600	3600
Grant	7000	2000



Trabajo con Fechas

- La base de datos Oracle almacena fechas en un formato numérico interno: siglo, año, mes, día, horas, minutos y segundos.
- El formato de visualización de fecha por defecto es DD-MON-RR.
 - Le permite especificar fechas del siglo XXI en el siglo XX especificando sólo los dos últimos dígitos del año
 - Le permite almacenar fechas del siglo XX en el siglo XXI de la misma manera

```
SELECT last_name, hire_date  
FROM employees  
WHERE hire_date < '01-FEB-88';
```

LAST_NAME	HIRE_DATE
King	17-JUN-87
Whalen	17-SEP-87



Trabajo con Fechas

SYSDATE es una función que devuelve:

- **Fecha**
- **Hora**

ORACLE

Copyright © 2004, Oracle. Todos los derechos reservados.

Aritmética de Fechas

- **Sume o reste un número a una fecha para obtener un valor de fecha resultante.**
- **Reste dos fechas para calcular el número de días entre dichas fechas.**
- **Sume horas a una fecha dividiendo el número de horas entre 24.**

ORACLE

Copyright © 2004, Oracle. Todos los derechos reservados.

Uso de Operadores Aritméticos con Fechas

```
SELECT last_name, (SYSDATE-hire_date)/7 AS WEEKS  
FROM employees  
WHERE department_id = 90 ;
```

LAST_NAME	WEEKS
King	744.245395
Kochhar	626.102538
De Haan	453.245395

ORACLE

Funciones de Fecha

Función	Resultado
MONTHS_BETWEEN	Número de meses entre dos fechas
ADD_MONTHS	Agrega meses de calendario a una fecha
NEXT_DAY	Día siguiente a la fecha especificada
LAST_DAY	Último día del mes
ROUND	Redondea la fecha
TRUNC	Trunca la fecha

ORACLE

Uso de Funciones de Fecha

Función	Resultado
MONTHS_BETWEEN ('01-SEP-95', '11-JAN-94')	19.6774194
ADD_MONTHS ('11-JAN-94', 6)	'11-JUL-94'
NEXT_DAY ('01-SEP-95', 'FRIDAY')	'08-SEP-95'
LAST_DAY ('01-FEB-95')	'28-FEB-95'

ORACLE

Uso de Funciones de Fecha

Supongamos que SYSDATE = '25-JUL-03':

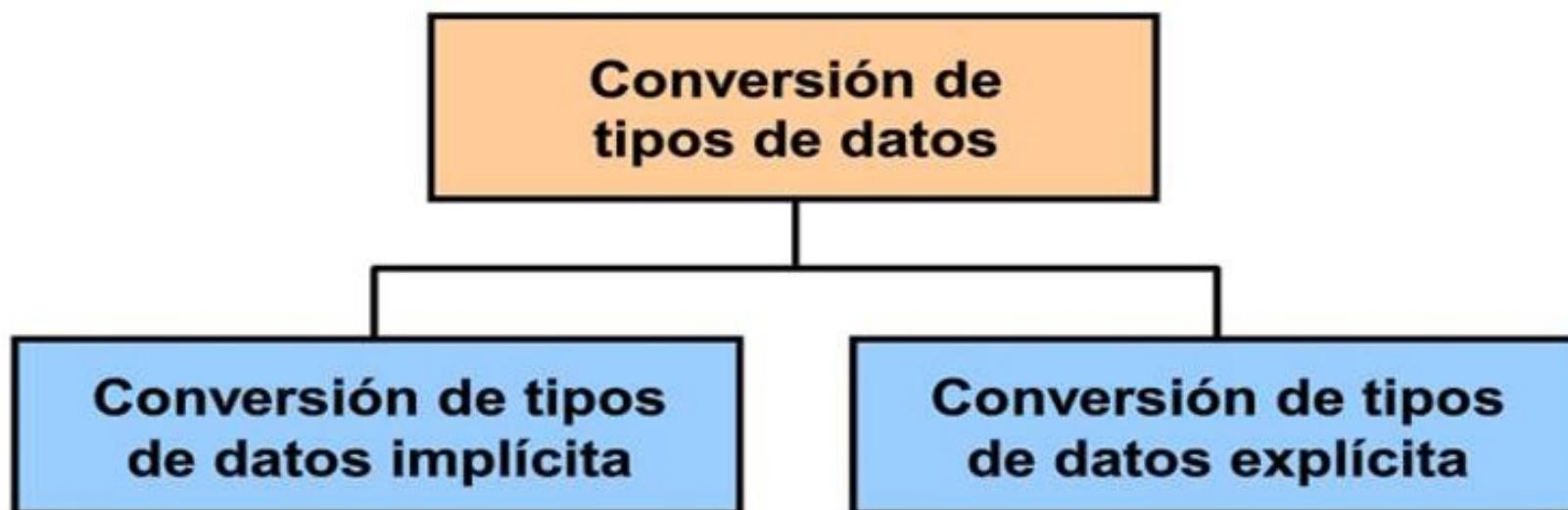
Función	Resultado
ROUND(SYSDATE, 'MONTH')	01-AUG-03
ROUND(SYSDATE , 'YEAR')	01-JAN-04
TRUNC(SYSDATE , 'MONTH')	01-JUL-03
TRUNC(SYSDATE , 'YEAR')	01-JAN-03

Práctica 3: Visión General de la Parte 1

Esta práctica cubre los temas siguientes:

- Escritura de una consulta que muestre la fecha actual
- Creación de consultas que requieren el uso de funciones numéricas, de carácter y de fecha
- Realización de cálculos de años y meses de servicio de un empleado

Funciones de Conversión



ORACLE

Conversión de Tipos de Datos Implícita

Para las asignaciones, Oracle Server puede convertir automáticamente:

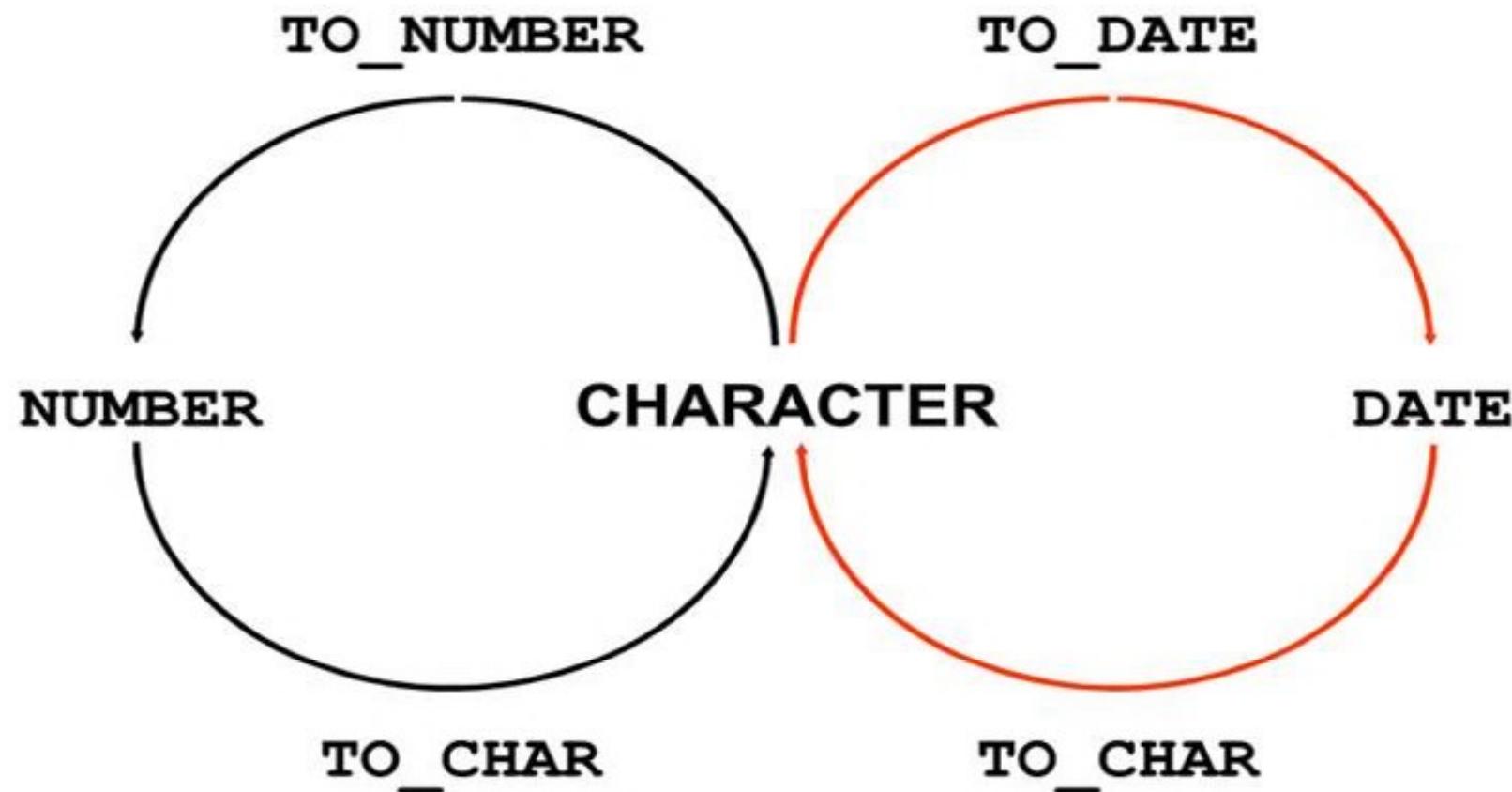
De	A
VARCHAR2 o CHAR	NUMBER
VARCHAR2 o CHAR	DATE
NUMBER	VARCHAR2
DATE	VARCHAR2

Conversión de Tipos de Datos Implícita

Para la evaluación de expresiones, Oracle Server puede convertir automáticamente:

De	A
VARCHAR2 o CHAR	NUMBER
VARCHAR2 o CHAR	DATE

Conversión de Tipos de Datos Explícita



ORACLE

Uso de la Función TO_CHAR con Fechas

```
TO_CHAR(date, 'format_model')
```

El modelo de formato:

- Debe ir entre comillas simples
- Es sensible a mayúsculas/minúsculas
- Puede incluir cualquier elemento de formato de fecha válido
- Tiene un elemento fm para eliminar espacios en blanco de relleno o suprimir ceros iniciales
- Está separado del valor de fecha por una coma

ORACLE

Elementos del Modelo de Formato de Fecha

Elemento	Resultado
YYYY	Año completo con números
YEAR	Nombre completo de año con letras (en inglés)
MM	Valor de dos dígitos para el mes
MONTH	Nombre completo del mes
MON	Abreviatura de tres letras del mes
DY	Abreviatura de tres letras del día de la semana
DAY	Nombre completo del día de la semana
DD	Día del mes con números

ORACLE

Copyright © 2004, Oracle. Todos los derechos reservados.

Elementos del Modelo de Formato de Fecha

- Los elementos horarios formatean la parte de hora de la fecha:

HH24 : MI : SS AM	15 : 45 : 32 PM
-------------------	-----------------

- Agregue cadenas de caracteres poniéndolas entre comillas dobles:

DD "of" MONTH	12 of OCTOBER
---------------	---------------

- Utilice sufijos para escribir el nombre completo de los números:

ddspth	fourteenth
--------	------------

ORACLE

Uso de la Función TO_CHAR con Fechas

```
SELECT last_name,  
       TO_CHAR(hire_date, 'fmDD Month YYYY')  
          AS HIREDATE  
FROM   employees;
```

LAST_NAME	HIREDATE
King	17 June 1987
Kochhar	21 September 1989
De Haan	13 January 1993
Hunold	3 January 1990
Ernst	21 May 1991
Lorentz	7 February 1999
Mourgos	16 November 1999

20 rows selected.

ORACLE

Copyright © 2004, Oracle. Todos los derechos reservados.

Uso de la Función TO_CHAR con Números

`TO_CHAR(number, 'format model')` ddspth

Éstos son algunos de los elementos de formato que se pueden utilizar con la función TO_CHAR para mostrar un valor numérico como carácter:

Elemento	Resultado
9	Representa un número
0	Muestra ceros
\$	Coloca un signo de dólar flotante
L	Utiliza el símbolo de divisa local flotante
.	Imprime un punto decimal
,	Imprime una coma como indicador de miles

ORACLE

Uso de la Función TO_CHAR con Números

```
SELECT TO_CHAR(salary, '$99,999.00') SALARY  
FROM   employees  
WHERE  last_name = 'Ernst';
```

SALARY
\$6,000.00

ORACLE

Copyright © 2004, Oracle. Todos los derechos reservados.

Uso de las Funciones TO_NUMBER y TO_DATE

- **Convierta una cadena de caracteres en formato numérico mediante la función TO_NUMBER:**

```
TO_NUMBER(char[, 'format_model'])
```

- **Convierta una cadena de caracteres en formato de fecha mediante la función TO_DATE:**

```
TO_DATE(char[, 'format_model'])
```

- **Estas funciones tienen un modificador `fx`. Este modificador especifica la correspondencia exacta del argumento de carácter y el modelo de formato de fecha de una función TO_DATE.**

ORACLE

Formato de Fecha RR

Año Actual	Fecha Especificada	Formato RR	Formato YY
1995	27-OCT-95	1995	1995
1995	27-OCT-17	2017	1917
2001	27-OCT-17	2017	2017
2001	27-OCT-95	1995	2095

Si los dos dígitos del año actual son:	Si los dos dígitos especificados del año son:		
	0–49	50–99	
Si los dos dígitos del año actual son:	0–49	La fecha devuelta está en el siglo actual	La fecha devuelta está en el siglo anterior al actual
	50–99	La fecha devuelta está en el siglo posterior al actual	La fecha devuelta está en el siglo actual

ORACLE®

Ejemplo de Formato de Fecha RR

Para encontrar empleados contratados antes de 1990, utilice el formato RR, que produce los mismos resultados si el comando se ejecuta en 1999 o ahora:

```
SELECT last_name, TO_CHAR(hire_date, 'DD-Mon-YYYY')
FROM employees
WHERE hire_date < TO_DATE('01-Jan-90', 'DD-Mon-RR');
```

LAST_NAME	TO_CHAR(HIR)
King	17-Jun-1987
Kochhar	21-Sep-1989
Whalen	17-Sep-1987

ORACLE

Anidamiento de Funciones

- Las funciones de una sola fila se pueden anidar hasta cualquier nivel.
- Las funciones anidadas se evalúan desde el nivel más profundo al menos profundo.

F3 (F2 (F1 (col,arg1) , arg2) , arg3)

Paso 1 = Resultado 1

Paso 2 = Resultado 2

Paso 3 = Resultado 3

ORACLE

Anidamiento de Funciones

```
SELECT last_name,  
       UPPER(CONCAT(SUBSTR (LAST_NAME, 1, 8), ' US'))  
FROM   employees  
WHERE  department_id = 60;
```

LAST_NAME	UPPER(CONCAT(SUBSTR(LAST_NAME,1,8))
Hunold	HUNOLD_US
Ernst	ERNST_US
Lorentz	LORENTZ_US

ORACLE

Funciones Generales

Estas funciones pueden utilizar cualquier tipo de datos y están relacionadas con el uso de valores nulos:

- NVL (`expr1, expr2`)
- NVL2 (`expr1, expr2, expr3`)
- NULLIF (`expr1, expr2`)
- COALESCE (`expr1, expr2, . . . , exprn`)

Función NVL

Convierte un valor nulo en un valor real:

- Los tipos de datos que se pueden utilizar son fecha, carácter y numérico.
- Los tipos de datos deben corresponder:
 - NVL(`commission_pct`, 0)
 - NVL(`hire_date`, '01-JAN-97')
 - NVL(`job_id`, 'No Job Yet')

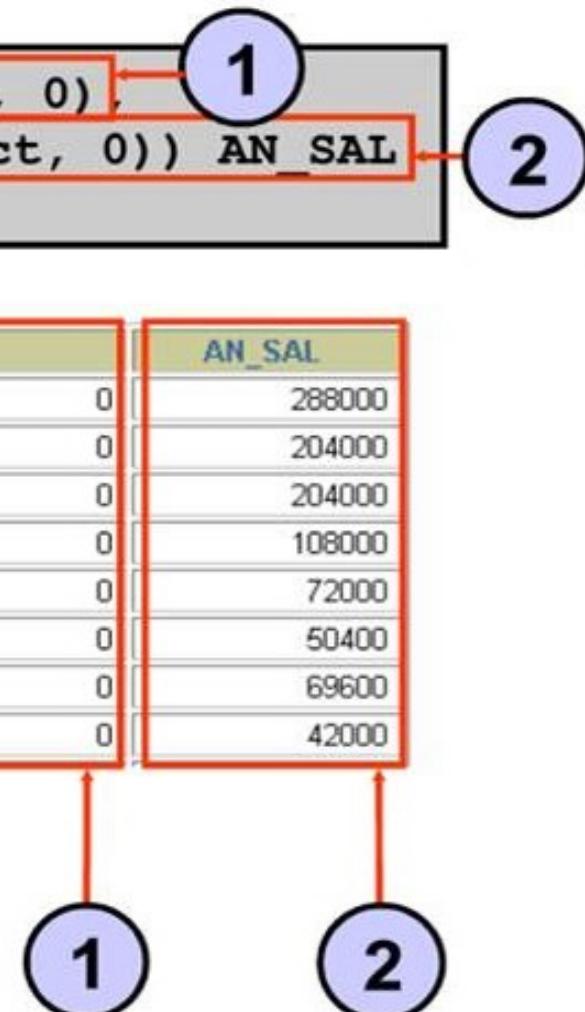


Uso de la Función NVL

```
SELECT last_name, salary, NVL(commission_pct, 0) 1  
      (salary*12) + (salary*12*NVL(commission_pct, 0)) AN SAL 2  
FROM employees;
```

LAST_NAME	SALARY	NVL(COMMISSION_PCT,0)	AN_SAL
King	24000	0	288000
Kochhar	17000	0	204000
De Haan	17000	0	204000
Hunold	9000	0	108000
Ernst	6000	0	72000
Lorentz	4200	0	50400
Mourgos	5800	0	69600
Rajs	3500	0	42000
...			

20 rows selected.



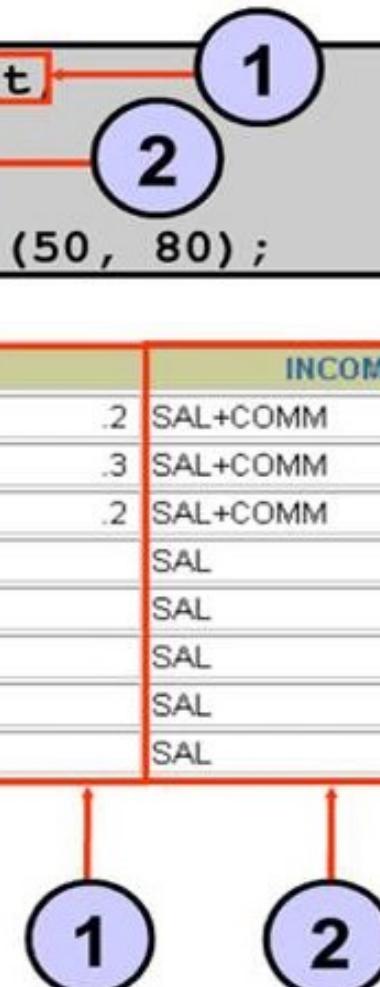
ORACLE

Uso de la Función NVL2

```
SELECT last_name, salary, commission_pct  
      NVL2(commission_pct,  
            'SAL+COMM', 'SAL') income  
FROM employees WHERE department id IN (50, 80);
```

LAST_NAME	SALARY	COMMISSION_PCT	INCOME
Zlotkey	10500	.2	SAL+COMM
Abel	11000	.3	SAL+COMM
Taylor	8600	.2	SAL+COMM
Mourgos	5800		SAL
Rajs	3500		SAL
Davies	3100		SAL
Matos	2600		SAL
Vargas	2500		SAL

8 rows selected.



ORACLE

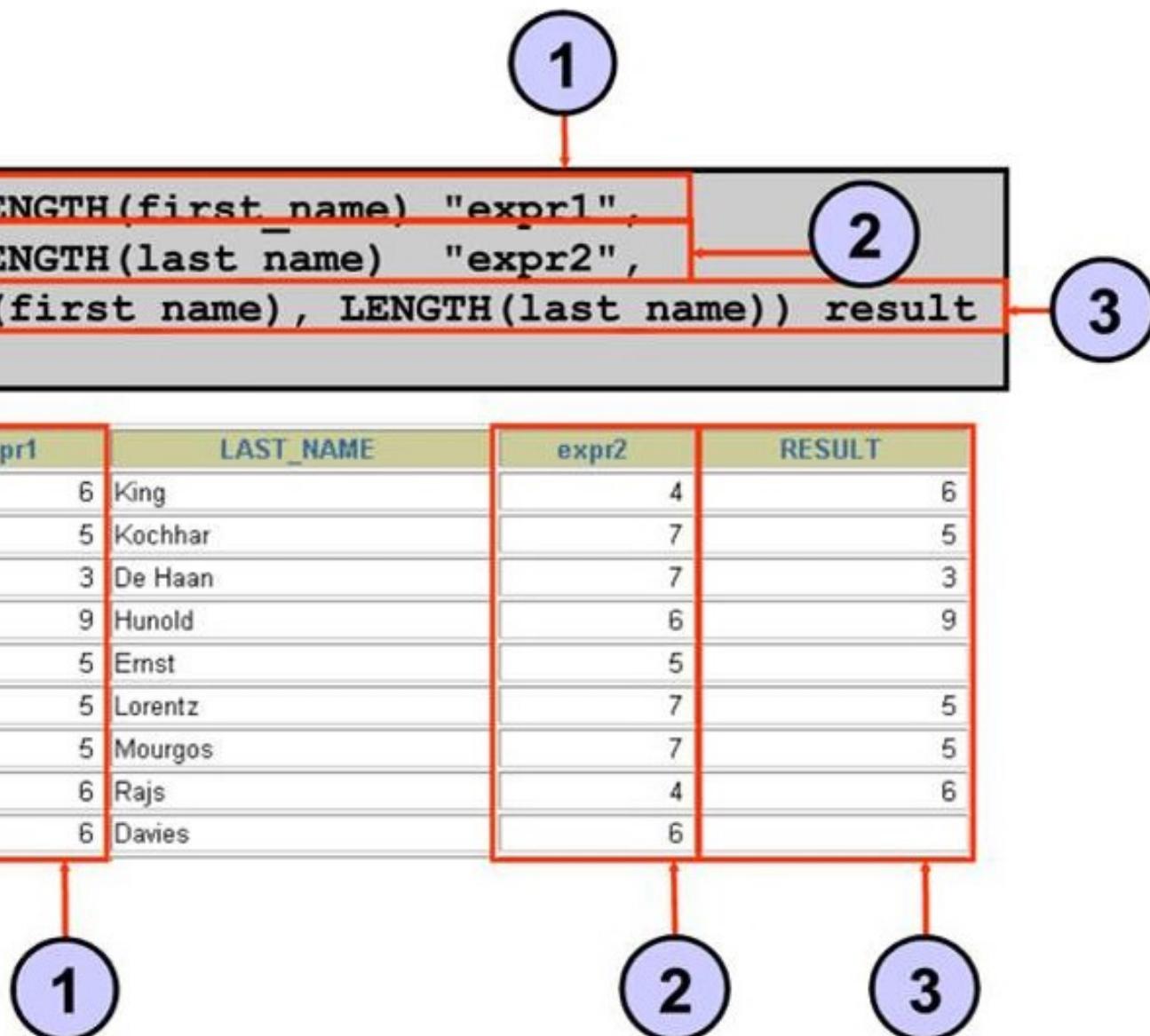
Uso de la Función NULLIF

```
SELECT first_name, LENGTH(first_name) "expr1",
       last_name, LENGTH(last_name) "expr2",
       NULLIF(LENGTH(first_name), LENGTH(last_name)) result
  FROM employees;
```

FIRST_NAME	expr1	LAST_NAME	expr2	RESULT
Steven	6	King	4	6
Neena	5	Kochhar	7	5
Lex	3	De Haan	7	3
Alexander	9	Hunold	6	9
Bruce	5	Ernst	5	
Diana	5	Lorentz	7	5
Kevin	5	Mourgos	7	5
Trenna	6	Rajs	4	6
Curtis	6	Davies	6	

...

20 rows selected.



ORACLE

Uso de la Función COALESCE

- La ventaja de la función COALESCE sobre la función NVL es que la primera puede tomar varios valores alternativos.
- Si la primera expresión no es nula, la función COALESCE devuelve esa expresión; en caso contrario, realiza una fusión (COALESCE) de las expresiones restantes.

ORACLE

Copyright © 2004, Oracle. Todos los derechos reservados.

Uso de la Función COALESCE

```
SELECT last_name,  
       COALESCE(manager_id,commission_pct, -1) comm  
FROM   employees  
ORDER BY commission_pct;
```

LAST_NAME	COMM
Grant	149
Zlotkey	100
Taylor	149
Abel	149
King	-1
Kochhar	100
De Haan	100

...

20 rows selected.

ORACLE

Copyright © 2004, Oracle. Todos los derechos reservados.

Expresiones Condicionales

- Permiten utilizar la lógica IF-THEN-ELSE dentro de una sentencia SQL
- Puede usar dos métodos:
 - Expresión CASE
 - Función DECODE

ORACLE

Expresión CASE

Facilita las consultas condicionales realizando el trabajo de una sentencia IF-THEN-ELSE:

```
CASE expr WHEN comparison_expr1 THEN return_expr1  
           [WHEN comparison_expr2 THEN return_expr2  
           WHEN comparison_exprn THEN return_exprn  
           ELSE else_expr]  
END
```



Uso de la Expresión CASE

Facilita las consultas condicionales realizando el trabajo de una sentencia IF-THEN-ELSE:

```
SELECT last_name, job_id, salary
      CASE job_id WHEN 'IT_PROG' THEN 1.10*salary
                    WHEN 'ST_CLERK' THEN 1.15*salary
                    WHEN 'SA REP' THEN 1.20*salary
      ELSE          salary END "REVISED SALARY"
FROM employees;
```

LAST_NAME	JOB_ID	SALARY	REVISED_SALARY
...			
Lorentz	IT_PROG	4200	4620
Mourgos	ST_MAN	5800	5800
Rajs	ST_CLERK	3500	4025
...			
Gietz	AC_ACCOUNT	8300	8300

20 rows selected.

ORACLE

Función DECODE

Facilita las consultas condicionales realizando el trabajo de una expresión CASE o de una sentencia IF-THEN-ELSE:

```
DECODE(col|expression, search1, result1
       [, search2, result2,...]
       [, default])
```

ORACLE

Copyright © 2004, Oracle. Todos los derechos reservados.

Uso de la Función DECODE

```
SELECT last_name, job_id, salary  
      , DECODE(job_id, 'IT_PROG', 1.10*salary,  
              'ST_CLERK', 1.15*salary,  
              'SA REP', 1.20*salary,  
              salary)  
      , REVISED_SALARY  
FROM employees;
```

LAST_NAME	JOB_ID	SALARY	REVISED_SALARY
Lorentz	IT_PROG	4200	4620
Mourgos	ST_MAN	5800	5800
Rajs	ST_CLERK	3500	4025
Gietz	AC_ACCOUNT	8300	8300

20 rows selected.

ORACLE

Uso de la Función DECODE

Muestre la tasa de impuestos aplicable para cada empleado del departamento 80:

```
SELECT last_name, salary,  
       DECODE (TRUNC(salary/2000, 0),  
                0, 0.00,  
                1, 0.09,  
                2, 0.20,  
                3, 0.30,  
                4, 0.40,  
                5, 0.42,  
                6, 0.44,  
                0.45) TAX_RATE  
  
FROM employees  
WHERE department_id = 80;
```

ORACLE

Resumen

En esta lección ha aprendido a:

- **Realizar cálculos en datos mediante funciones**
- **Modificar elementos de datos individuales mediante funciones**
- **Manipular la salida para grupos de filas mediante funciones**
- **Modificar formatos de fecha para su visualización mediante funciones**
- **Convertir tipos de datos de columnas mediante funciones**
- **Utilizar funciones NVL**
- **Utilizar la lógica IF-THEN-ELSE**



Práctica 3: Visión General de la Parte 2

Esta práctica cubre los temas siguientes:

- **Creación de consultas que requieren el uso de funciones numéricas, de carácter y de fecha**
- **Uso de concatenación con funciones**
- **Escritura de consultas sensibles a mayúsculas/minúsculas para probar la utilidad de las funciones de carácter**
- **Realización de cálculos de años y meses de servicio de un empleado**
- **Determinación de la fecha de revisión para un empleado**